

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ - Campus de Curitiba (Brasil) -
Departamento Acadêmico de Eletrônica (DAELN). Disciplina:
Fundamentos de Programação 2 - IF62C / Turma: S11/S12. Prof: Jean M. Simão.
Curso: **Eng. Eletrônica. Prova sobre Orientação a Objetos – 2ª Parcial.**

Nome do Aluno: _____ Turma: _____
Horário de Começo: _____ Horário de Fim: _____

**Leia toda a prova, pois os enunciados estão completados uns nos outros.
Utilize os bons princípios de projeto e programação orientada a objetos.**

(Questão - 1) – (a) Em um programa C++ (para *console*), crie uma classe chamada *Nave* com um atributo privado inteiro chamado *número*, certamente com seus respectivos métodos *get* e *set*. Esta classe terá um único método virtual chamado “*virtual void informe()*” que informará em tela o valor de número. Outrossim, esta classe poderá ter outros métodos que não sejam virtuais, incluindo construtora e destrutora. Ainda, quando pertinente, usar elementos constantes (i.e. atributos, funções, retornos e parâmetros constantes) nesta classe, assim como nas demais classes solicitadas.
– **(b)** Sobrecarregue o operador de recebimento (*operator=*) na classe *Nave* de maneira que qualquer objeto baseado nela possa receber um valor para o atributo *número* por este meio.

(Questão - 2) Crie uma classe *Lista_Naves* (em C++) para apontamento de objetos de *Nave* baseando-se na classe *list* da *STL*. Entretanto, na prática, esta lista será útil para conter apontamento de objetos de subclasses de *Nave* em um quadro de polimorfismo. Ainda, a classe *Lista_Naves* deverá ter um método para incluir, um método para excluir e outro para listar. Particularmente, a cada chamada, o método incluir permitirá incluir (um apontamento) sempre na última posição da lista. Por sua vez, o método excluir permitirá excluir (um apontamento) de qualquer posição da lista conforme parâmetro (de posição na lista) dado na chamada do método. Por fim, o método listar chamará cada método *informe* de cada objeto (referenciado na lista) utilizando-se de um iterador.

(Questão - 3) Em C++, crie uma classe *Avião* (com atributo privado booleano *supersônico* e seus respectivos *get* e *set*) e uma classe *Navio* (com um atributo privado inteiro *capacidade* e seus respectivos *get* e *set*), ambas derivadas de *Nave* e cada qual com um método chamado “*void informe()*”. No caso de *Avião* tal função-membro informará (em tela) “*Transporte Aéreo*”, se é supersônico ou não e o valor do atributo *número*. No caso de *Navio*, por sua vez, informará “*Aquático*”, a *capacidade* e o valor do atributo *número*.

(Questão - 4) Crie uma classe *Principal* cujo único objeto poderá armazenar em si ponteiros de objetos *Avião* por meio de um método *void incluir_Avião (Avião* pa)*, bem como poderá armazenar em si ponteiros de objetos *Navio* por meio de um método *void incluir_Navio (Navio* pn)*. Entretanto, estes apontamentos de objetos *Avião* e *Navio* serão armazenados como apontamentos de objetos *Nave*, por meio de *static_cast*, em um objeto agregado instanciado a partir da *Lista_Naves*. Ainda, faça com que a classe *Principal* agregue objetos da classe *Avião* e objetos da classe *Navio* cujos endereços serão armazenados como apontamentos, no objeto da classe *Lista_Naves*, por meio de chamadas apropriadas de *incluir_Avião* e *incluir_Navio* na construtora de *Principal*. Por fim, haverá um método *executar* na *Principal* que chamará o método listar de tal objeto lista.

(Questão - 5) - (a) Elabore um Diagrama de Classes (em *UML*) que modele as classes requisitadas nas questões precedentes, bem como seus relacionamentos.
- **(b)** Explique a importância de Padrões de Projeto? Cite cinco padrões de Projeto? Responda se foi utilizado ou não algum padrão de projeto nas questões anteriores, citando qual padrão de projeto se sim.